IAC-06-D1.4.08

EXECUTIVE MODULAR CONTROL OF HETEROGENEOUS SPACECRAFT COMPONENTS AND AGENTS

B. J. Glass[1], H. Cannon[1], S. Christa[1], S. Huffman[1] and J. Johnson[2]
[1]NASA-Ames Research Center, Moffett Field, CA 94035 USA ; [2] US Army Cold Regions Research Laboratory, Ft. Wainwright, AK 99703, USA;
brian.glass@nasa.gov

Abstract

Future lunar and Mars exploration missions will incorporate legacy software and hardware components from the then-existing state of the art. Integrating the command, control, and data from complex networks of heterogeneous instruments, components, agents and human operators requires a modular, plug-and-play architecture. One example of such an approach has been developed for subsurface prototype missions, with multiple clients, servers, and platform support provided by a publish/subscribe architecture that combines rover-developed executive software control with a middleware layer based on a CORBA subset. This architecture has been demonstrated in several successful field tests (automated drilling tests in Spain, the US and Arctic Canada) in 2005-2006 to support the dynamic addition of and deactivation of spacecraft platform components without requiring recoding or patching.

Introduction

Since the dawn of the Space Age… no new mission or spacecraft is a completely clean-sheet design, unconstrained by the reuse or adaptation of previously-designed components. Future missions and spacecraft will build on the experience and hardware and software base created previously. As vehicle system health management becomes more commonplace, and as instruments are increasingly designed with their own operating software agents and using existing data formats, future mission and spacecraft designers will be left with a choice between integrating multiple heterogeneous instruments and multiple smart-spacecraft components, or redesigning these. Mission cost constraints tend to favor the former whenever technically feasible.

For quick, low-cost mission prototypes, it is even more necessary to bring together legacy hardware and software components. Faced with a need to command, control and acquire data from an assemblage of existing instruments and components for several planetary subsurface prototype development projects, a team at NASA Ames adapted best-commercial-practices software, together with a software executive developed for rover control, to quickly and flexibly integrate otherwise-incompatible off-the-shelf instruments and components. The Mars Analog Río Tinto Experiment (MARTE) Instrument Interface (MInI)[1] is a communications package for passing commands between multiple clients and servers. MInI is an advanced version of a previously developed piece of software called K9Client[2]. Designed for the communications between the control software and the spacecraft platform on the

MARTE project, MInI makes use of the K9Client interface to pass strings of characters across multiple platforms and eventually across multiple languages. MInI uses the Common Object Request Brokering Architecture (CORBA) to accomplish this multiple platform support. Whereas K9Client was designed to communicate with a single server, MInI extends that capability to support multiple servers on independent machines.

The Construction and Resource Untilization Explorer (CRUX) robotic lunar exploration concept was comprised of legacy instruments and spacecraft subsystems (many flown on earlier missions) as well as an automated drill. The CRUX Executive Controller (CEC) provided the integration, top-level control, and process communication needed to link the components and software agents of CRUX together into an operational drilling and survcying prototype.

The core functionality of the CRUX Executive Controller (CEC) is its ability to integrate and control heterogeneous instruments and the CRUX platform subsystems. This capability is built upon the MARTE Instrument Interface (MInI), which is a simple and flexible communications package that was originally developed to ease the software development and integration process for the Mars Astrobiology Research and Technology Experiment (MARTE). MInI has supported the development of many instruments and control systems across a number of widely separated institutions in Spain, Texas, Callfornia, Oklahoma, and New York. These mission pieces needed to be developed independently at separate home institutions, but yet come together during a short integration period and communicate across a number of different computing and control platforms. MInI was developed in order to facilitate this process, and is the basis for the CEC. The MInI software approach was in turn adapted from a small subset of the Common Object Request Brokering Architecture (CORBA), thus enabling it to communicate seamlessly across a wide range of platforms and operating systems. MInI was a publish/subscribe architecture, and allowed any number of clients to connect to any number of servers.

The architecture and MInI tools on which the CEC was based underwent several field tests in 2005-6. In June 2005 the Executive and a Honeybee drill and several instruments and a core handling system were integrated and tested during drilling tests in a quarry in Santa Cruz, California. The same set was shipped to Rio Tinto, Spain and integrated with a Spanish-developed Borehole Inspection System probe prior to more drilling tests there in late 2005. [3,4] The CEC components were also tested successfully in daily use in permafrost drilling with a separate Honeybee drill and with realtime fault diagnosis and recovery in the Arctic (Haughton Crater, Devon Island, Canada) in July 2005 and July 2006.[5]

This paper will look at the CRUX example as a motivation for modular integration in future missions, then discuss the MInI and TInI architectures. Recent field test results will be given from the DAME and MARTE projects which used MInI, followed by conclusions.

Original Motivation and Goals for CRUX

The CRUX project was undertaken to develop critical technology needed to identify optimal sites to conduct lunar and planetary surface operations (LPSO) related to *in-situ* resource utilization (ISRU), construction, environmental management, and surface mobility. Understanding the "lay of the land", the available resources, and geotechnical properties is critical to the success of future lunar and planetary missions. Successful LPSO will require a good knowledge of surface topography, geotechnical properties (e.g., grain size, mineralogy, bulk density, thermal and mechanical properties) and whether water is present. CRUX goals were to develop an integrated, but modular, suite of instruments and related software, referred to collectively as the Prospector-Surveyor/Mapper-Decision Support System (P-S/Mapper-DSS) to fulfill these requirements. The system consisted of an instrumented drill (Prospector), surface geophysical and optical mobile sensors (Surveyor), linked with a mapping and decision support system

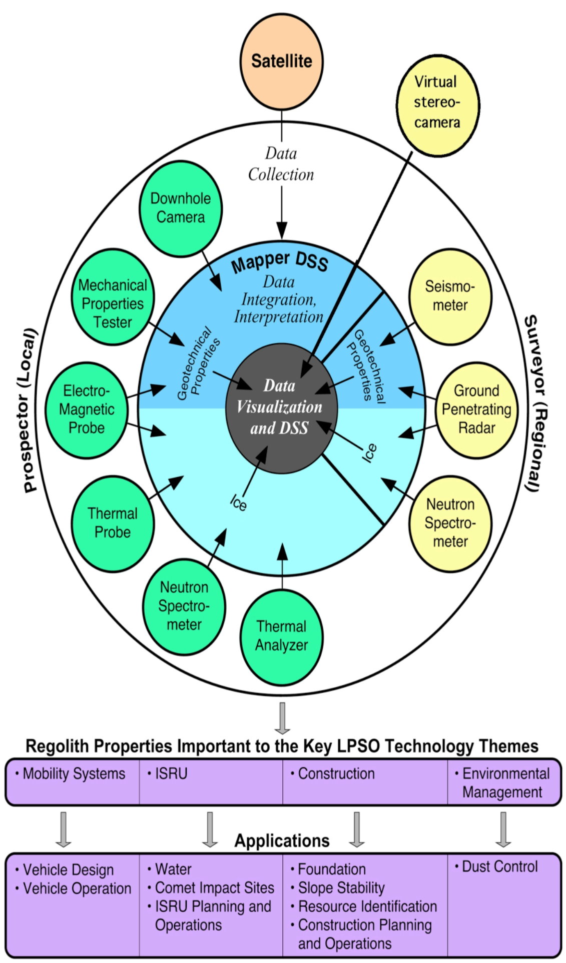

Figure 1. CRUX Prospector-Surveyor data interdependencies .

(Mapper-DSS). The goal of instrument suite modularity was to allow instruments to be added, removed, or reconfigured according to mission requirements without affecting the performance of remaining instruments. The Mapper-DSS would absorb data from satellites, historical records and the Prospector-Surveyor data (the Mapper-DSS would automatically adjust to the instrument suite configuration). Satellite data was used to identify a target of interest, at coarse resolution. The Surveyor's geophysical instruments would map shallow subsurface regions to help locate optimal drilling sites. The Prospector's drill would carry instruments down-hole to measure site-specific regolith geotechnical properties and detect water to a depth of 2 m. When linked to borehole data, Surveyor data could provide a way to map geotechnical and ISRU properties between boreholes. Stereo pair images were to be used to generate regional topography.

While CRUX instrument measurements would be useful individually, their true value would lie in their integration and fusion using the Mapper/DSS. For example, the neutron spectrometer can be used to determine the quantity of hydrogen, but that hydrogen may be elemental, water ice, or some other hydrogen-rich compound. By combining information from the Thermal Analyzer (TA), which can directly determine the presence of water, it would become possible to estimate the percentage of hydrogen associated with water. The local water-to-hydrogen percentage can be used to estimate the regional partition of hydrogen between water and other compounds using data from the surface mounted neutron probe. Thus, the instruments of Prospector-Surveyor are complementary and their data could be used in Mapper-DSS to derive three-dimensional maps of the geotechnical and water data necessary for LPSO.

In CRUX, the Prospector-Surveyor instruments largely exist, or exist as high-maturity prototypes. Integrating these on Prospector and Surveyor, as well as getting their integrated data to the Mapper/DSS, required executive command and control software as well as some way to communicate with a variety of existing components.

Modular Approach to Instrument Integration

CEC Requirements

Command, control, and data integration for the CRUX instruments is provided by the CRUX Executive Controller instrument integration component (CEC). The Executive Controller was required to provide the integration, top-level control, and interprocess communication needed to link otherwise-incompatible legacy instruments and components together into an operational lunar drilling and prospecting prototype. In addition to spacecraft-level instrument control, the CEC was required to provide the software "glue" for system integration and a user interface for mission operations tests. Earlier versions had been field tested on other projects with prototype instruments and drills at relevant sites (Spain and the Arctic). Just as importantly, its ability to quickly and reliably integrate different instruments and subsystems enabled their testing together -- and hence raising the technology maturity level of other instruments and spacecraft components, and of a given lander or rover platform as a whole.

Instrument Integration with CORBA (MInI)

CEC is based on the MARTE Instrument Interface (MInI) which is a communications package that was developed in order to facilitate the integration of a number of instruments and devices for the MARTE project. The MARTE project was a complex project that brought together subsystems and instruments that were developed at NASA Ames, NASA JSC, the Center for Astrobiology in Spain, Honeybee Robotics in New York, and the University of Oklahoma, as well as a number of commercial instruments. Rather than restrict the institutions or selected instruments to work on a common platform and operating system, MInI instead provided a interface layer that would allow systems on a wide variety of platforms to communicate easily.

MInI is an advanced version of a previously developed piece of software called K9Client[1], which was designed to communicate between the control software and the rover platform on the MARTE project. It makes use of the K9Client interface to pass strings of characters across multiple platforms and eventually across multiple languages. Whereas K9Client was designed to communicate with a single server, MInI extends that capability to support multiple servers on independent machines. MInI also adds increased support for new server creation and utilities for working with K9Client (including a platform-independent graphical user interface). The design for MInI is shown in Figure 2. The interface makes use of K9Client as a part of its implementation, rather than reconstructing the functionality of K9Client.

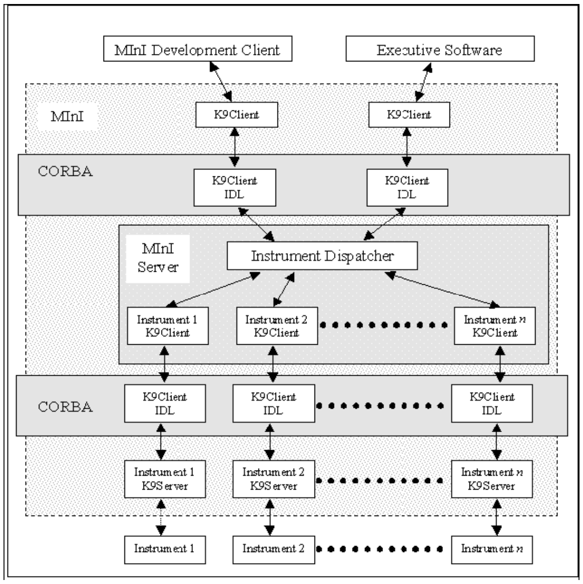

Figure 2. Design for the MARTE Instrument Interface (MInI).

TInI

The Tiny Instrument Interface (TInI) is a slimmer version of the MARTE Instrument Interface (MInI) with the same functionality. Since the type of data passed varied from command to command, strings were chosen as an interface where any such data could be encoded. This method also allowed talking to servers using telnet clients, which any operating system comes with one.

The MARTE team developed MInI to allow the K9client interface to work with multiple clients and servers, to provide broadcast capability, and to allow for faster server design. While all of these changes made K9Client better, they did not make it faster. Because the DAME project needed an extremely lightweight version of MInI, to support high-speed data transmission from the drill, DAME developed TInI. TInI removed the largest and most cumbersome portion of MInI—the ACE+TAO CORBA interface—and replaced it with a socket interface. The CORBA transport option was then added back to TInI at a later date, to permit more flexibility.

While the original goal of TInI was to provide an interface between the control software and the multiple instruments on the DAME platform, there are many other potential applications for TInI. Just as K9Client provides a facility for passing strings between any two servers, so does TInI provide the facility for passing strings between any number of servers. Because of the extensible model of TInI, wherein server names are specified in the command, the interface can be extended to new applications merely by writing new servers and clients. Using TInI requires no special knowledge about a particular server aside from its name—all other relevant information can be obtained programmatically.

Rover-derived Executive

The Contingent Executive was originally developed at NASA Ames Research Center to control planetary rovers. It was tested extensively onboard NASA Ames' Marsokhod rover and the K9 Rover during numerous field tests occurring between 1999 and 2003 [1,6]. It was also modified and used to control the drill and onboard science instruments for the Mars Astrobiology Research and Technology Experiment (MARTE) field test 2005 [4].

The Contingent Executive uses a plan language known as the Contingent Rover Language (CRL) to serve as the communication medium for receiving instructions from the ground operations team. A CRL plan contains a sequence of tasks to be executed along with temporal and state conditions that must be met before, during, and after each task executes. A CRL plan may also contain branches, which allow different plan segments to be run based upon the conditions that are encountered at run time. The baseline plan is normally executed as specified, but may be interrupted by the insertion or replacement of an alternate plan (ie. recovery procedure). These are a separate set of plans which are kept in memory, and execute when a set of eligibility conditions are met.

MARTE Integration – example

Because MInI and the Executive system use standard network and operating system platforms, they can be used to meet a wide range of needs with regards to Remote Operations. The client/server architecture used by MInI allows for integration of instruments in the same room, or across an ocean. It also allows access to the systems by remote users – either the Executive or the instruments themselves.

In previous field tests, the system has also been integrated with a remote web based Science Data Browser, used by remote science teams to view data collected by instruments, discuss that data, and make suggestions for future work to the Mission Operations team based on their findings. This was made easier by the amount of uniform data provided by this system. Every data file created has an accompanying XML file listing times, dates, depths or any other critical information needed for interpretation. A common naming scheme also makes this process easier.

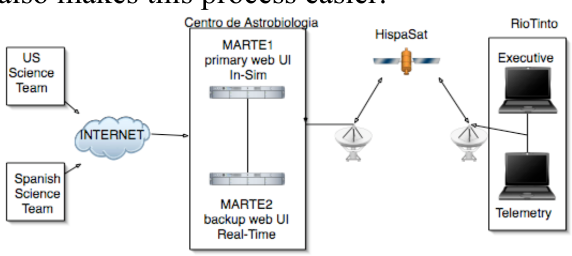

Figure 3. MARTE Remote Operations architecture incorporating MInI.

Figure 4 shows the architecture for the MARTE project. To simplify data communications, we took a centralized approach to command and data handling. That is, all commands to each subsystem came only from the executive (except in the case of human intervention, where commands could be sent via a client GUI). All data that was generated by the instruments was stored in a site science repository. The data in the site repository was accessed by the telemetry interface for transmission to the mission operations center.

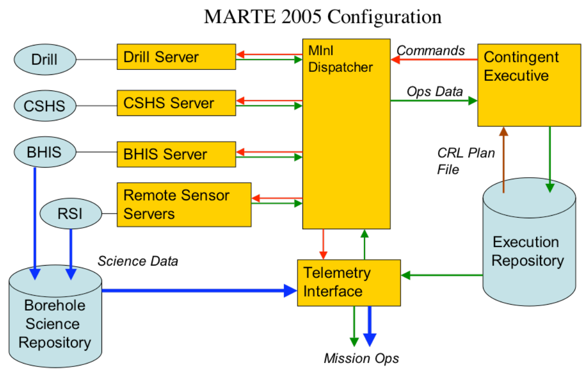

Figure 4. MARTE 2005 Configuration.

For the MARTE project, the Contingent Executive read in a plan file from the Execution Repository that was generated from scientist inputs. The plan specified what each subsystem should do, and in what order they should be accomplished. The Executive then sent commands to each of the subsystems via MInI, and received back status information so the task could be monitored. For instance in a typical drilling sequence, the Executive would first send a command to the drilling subsystem to drill a core. It would then instruct the drill to raise the core out of the ground and into the core transfer device, while concurrently it asked the core sample handling subsystem to retrieve an empty clamp and move it to the core loading area. Then it would command the core sample handling subsystem to move to each instrument location, and subsequently cause each instrument to take a measurement. Finally, after all of the data was stored, the Executive would command the telemetry interface to fetch the data, and transmit it to mission operations.

Several other sequences were available to the scientists to choose from, such as a life detection sequence (subsampling the core, crushing, and insertion into a life detection instrument), borehole inspection via a probe, retrieval of stored core for additional measurements, etc. By providing these sequences as plan files, the sequences could easily be modified without changing code to fit the current situation.

For the DAME project, the Contingent Executive was used to control the drill system as well as to accept state estimates from multiple diagnostic systems through the TInI interface. The baseline drill plan contained commands to move to the bottom of the hole, drill a fixed distance, and then pull up off the bottom and wait in order to take a temperature measurement. This was repeated for a set number of times. During these operations, the executive would combine the inputs from the diagnostic systems. If the combined inputs pointed to an ongoing problem, the Executive would insert a recovery procedure. After the recovery procedure executed, the Executive would continue where it left off with the baseline plan.

Field Test Results

MARTE drilling and life detection

MARTE performed a simulation of a Mars drilling mission in September 2005 including interpretation of drill mission results by a remote science team in a blind test. The field experiment was an unqualified success. The MARTE lander was placed near the site of the microbiology drilling campaign at Rio Tinto, Spain. Science team participants included members of the planetary geology and astrobiology community. Science teams located at CAB in Madrid and at Ames in California commanded the mission operation for two weeks each. During the mission simulation, the drilling achieved a depth of 6 meters into a weathered gossan deposit. Average core recovery was 20% in this unconsolidated material. Borehole inspection imaging and spectroscopic measurements of the hole walls supplemented the incomplete core record. Using the combination of instruments, the science team was able to correctly identify the geologic nature of the site, correctly interpreted the mineralogy, and selected sites for life detection experiments that yielded positive (for life) results.

The onboard subsystem automation through the use of MInI and the Contingent Executive worked very well, and helped to streamline operations at the borehole. Initial concerns proved groundless, that by trying to add automation to the project, the overall field test would have difficulty reaching its depth target because of the limited time for integration and system verification. Actual experience was that the sequences (drilling, sample handling, core processing, science measurements) required literally hundreds of commands to be executed. The drilling and science goals could not have been achieved without the onboard automation. Likewise, by integrating the automation elements with the remote operations infrastructure, the science team was able to receive timely-enough information to enable them to specify plans for each work day.

DAME hands-off drilling success

This modular approach to instrument integration also showed its adaptability in the Drilling Automation for Mars Exploration (DAME) project, where MInI (later TInI) was used to integrate three separate drilling-diagnostic software modules, each of which operated in parallel with drilling operations. DAME developed and tested standalone automation at a lunar/martian impact crater analog site (Haughton Crater Research Station, Devon Island, Nunavut) in Arctic Canada. (HCRS) Figure 5 shows the DAME drill during these field tests.

Figure 5. DAME Automated Drilling Tests on Devon Island

The search for resources and past/present life on other planetary bodies will require subsurface access, which requires exploratory drilling. Drilling has been a hard, human-intensive problem in terrestrial applications, but planetary drills require automation. DAME's overall goal was to develop and test a capability for hands-off, unmonitored drilling operations, including responding to changing drilling conditions and strata. Together with the drill-string changeout and core-handling automation demonstrated by its sister MARTE project, DAME demonstrated the comprehensive remote control and management of science drilling that is required for future subsurface access to other planets. This capability gains credibility from its validation and testing outside the laboratory at a remote Mars-analog site.

Figure 6 shows the 2006 DAME software architecture. [7] All of the data transferred between the modules in this architecture used the Tiny Instrument Interface (TInI) communications backbone.

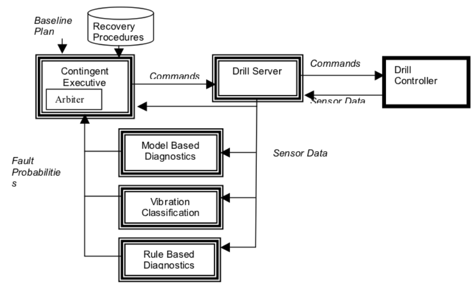

Figure 6. DAME Software Architecture

There were three sets of 2006 DAME test goals. The first was to demonstrate the automated recognition, while drilling, of at least three of the six major fault modes for the DAME drill, and to employ the correct recovery or safing procedure in response. Any faults not seen naturally in the course of drilling would be manually induced at the end of testing. The second set of 2006 goals was to operate for three or more hours autonomously, hands-off. And the third 2006 goal was to exceed 3m into the permafrost with the DAME drill (it had not gone further than 2.2m previously). And ground truth drilling would use small commercial drilling equipment in parallel in order to obtain cores and ice profiles from the permafrost. All three DAME 2006 test goals were completed successfully. All six faults were encountered naturally in the course of drilling, none had to be artificially induced, and the last of the six occurred on 24 July, a week into drilling. Five of the six faults were correctly identified, repeatedly, corrective actions were taken by the automation software and drill, and drilling continued. 44 hours of hands-off drilling, under executive software control, were logged – limited by the need to shut down power periodically to refill the generator at the remote drilling site.

Once all DAME field test goals had been met, an extended, higher-risk test series was conducted during the last two days in the field in July 2006. Previous hands-off tests had been run with humans nearby, monitoring drilling progress and software responses at the drill site, ready to intervene in order to save the equipment in case of a general system failure.

A “bare” or “exposed” test was run on the evening of 27 July. This consisted of starting an automated drilling sequence, and then directing the human staff to leave the equipment completely unattended while having dinner several miles away at the H base camp. This caused some nervousness among the programmers and engineers, but was a success -- as upon their return four hours later. the automated sequence was still going on and the DAME system had detected and successfully responded to a fault and continued on.

A remote test was run early on 28 July, initiating a drilling sequence and monitoring the progress remotely via the data link from the crater floor to the HCRS base. Remote “uplink” and “downlink” of drilling data and commands was not a DAME project requirement, but will be necessary for a flight instrument and is supported by the executive and MInI/TInI architecture used here.

Conclusions

The flexibility added by a disintermediation layer – the MInI/TInI spacecraft middleware – allowed reconfiguration of instruments, hardware and software modules dynamically without requiring recoding and validation for every configuration change. This approach decouples each robotic (and eventually, human) network element from a need to know each other’s internal state or data, and it integrates them in a software-bus architecture. Each becomes a black-box in the view of the others in a broader spacecraft-wide or planetary tactical surface network. This approach has been successful in remote Earth-based drilling and instrument coordination and control, tested at several field sites. While the original goal of MInI and TInI development was to provide an interface between the control software and the multiple instruments on the MARTE platform, there are many other potential applications. MInI/TInI provides the facility for passing strings between any number of onboard or tactical-area servers. Because server names are specified in MInI/TInI commands, this interface can be extended to new applications merely by writing new servers

.

On top of the MInI/TInI layer was executive-reasoning software, which executed plans and sequences and negotiated spacecraft resources between competing instruments and spacecraft subsystems. The combination of a generic executive with a publish/subscribe architecture allowed on-the-fly additions and subtractions of data sources and command recipients – spacecraft plug-and-play.

The CEC architecture was based upon modest previous automated-spacecraft-drilling projects (DAME and MARTE) and rover executive software. Eventually, humans will participate remotely by supervising these semi-autonomous (toddler-like, in some sense) instruments, rovers, drills or other software agents on an intervention-as-needed basis in mission operations. Larger-scale, more heterogeneous tests that include both humans in spacesuits and in remote mission operations along with robotic components will likely need even more flexibility and plug-and-play redeployment capabilities. Future exploration systems technology field tests could be used to piggyback the validation of this executive modular software architecture.

Acknowledgements

The MARTE project is part of the NASA Astrobiology Science and Technology for Exploring Planets (ASTEP) program. The DAME project is part of the NASA Mars Instrument Development Program (MIDP). Thanks to Carl Pilcher, Karen McBride, Michael Meyer and David Lavery at NASA Headquarters for their support of these efforts. The authors also thank their team members at Honeybee Robotics, Georgia Tech, the Centro de Astrobiologia and colleagues in the US and Spain working on the DAME and MARTE projects.

References

1. "MInI User's Guide," H. Cannon and M. Branson, Version 1.0, Computational Sciences Division, NASA-Ames Research Center, Moffett Field, CA, March 2005.

2. Bresina, J. L, et al., "Increased flexibility and robustness of Mars rovers", *Proceedings of i-SAIRAS '99, The 5th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, 1999.

3. Stoker, C., et al., "Mars Analog Río Tinto Experiment (MARTE): 2003 Drilling Campaign To Search For A Subsurface Biosphere At Río Tinto, Spain," *35th LPSC*, Abstract 2025, 2004.

4. Stoker, C., et al., "Field Simulation Of A Drilling Mission To Mars To Search For Subsurface Life," *36th LPSC*, Abstract 1537, 2005.

5. Glass, B. et al., "Drilling Automation For Subsurface Planetary Exploration," *Proceedings of 8th iSAIRAS*, Munich, Germany, September 2005.

6. Pederson, L., Bualat, M., Smith, D.E. ,and Washington, R.,"Integrated Demonstration of Instrument Placement, Robust Execution and Contingent Planning", *iSAIRAS 2003,* Nara, Japan, May 2003.

7. Glass, B., et al., "DAME: Planetary-Prototype Drilling Automation," *AIAA Space 2006*, San Jose, California, Sept. 2006.